

[首页](#)[官方教程](#)[视频课程](#)[社区开源](#)[正在开发中.....](#)[登录](#)

Spring Data JPA系列：投影（Projection）的用法

在JPA的查询中，有一个不方便的地方，`@Query`注解，如果查询直接是`select c from Customer c`，这时候，查询的返回对象就是`Customer`这个完整的对象，包含所有字段，对于我们的示例并没有什么问题，但是对于比较庞大的domain类，这个查询时就比较要命，并不是所有的字段都能用到，比较头疼。另外，如果定义`select c.firstName as firstName, c.lastName as lastName from Customer c`这个查询结果，返回的对象是`Object`类型，而且无法直接转换成`Customer`对象，这样用起来就不是很方便。对于这种情况，JPA提供了一种声明方式来解决，即声明一个接口类，然后直接使用这个接口类接受返回的数据即可。下面奉上代码：

1、增加CustomerProjection接口类

```
package com.example.demo.dto;
import org.springframework.beans.factory.annotation.Value;
/**
 * Created by Administrator on 2017/7/9 0009.
 */
public interface CustomerProjection {
    @Value("#{target.firstName + ' ' + target.lastName}")
    String getFullName();

    String getFirstName();

    String getLastName();
}
```

这里声明的方式是可以直接通过`get+属性名`，这是普通的，另外也可以通过`@Value`注解来实现

2、增加CustomerRepository方法

```
@Query("SELECT c.firstName as firstName, c.lastName as lastName fr
Collection<CustomerProjection> findAllProjectedBy();
```

3、增加CustomerController方法

```
/**
 * find by projections
 */
@RequestMapping("/findAllProjections")
public void findAllProjections(){
    Collection<CustomerProjection> projections = repository.findAllProjectedBy();
    System.out.println(projections);
    System.out.println(projections.size());
    for (CustomerProjection projection:projections){
        System.out.println("FullName:"+projection.getFullName());
        System.out.println("FirstName:"+projection.getFirstName());
        System.out.println("LastName:"+projection.getLastName());
    }
}
```

这里只是做了简单示意，深入的内容需要自己去挖掘探索。不过关于Projection的资料比较少，我也是扒了不少资料才理解的差不多了，还需要多多实践。

另外spring-data-examples项目中有一些JPA的例子，可以用来学习，梳理思路。

<https://github.com/spring-projects/spring-data-examples/tree/master/jpa>

[Spring Data](#)

阅读	评论	喜欢
286	2	0

[喜欢](#)[SpringForAll](#)

文章	问答	粉丝
79	6	236

[关注](#)